MITIGATION OF DENIAL OF SERVICE ATTACKS IN SOFTWARE-DEFINED-COGNITIVE RADIO NETWORKS USING SOFTWARE-DEFINED-COGNITIVE RADIO SHIELD

LEBEPE MAMPUELE

DISSERTATION

Submitted in fulfillment of the requirements for the degree of

MASTER OF SCIENCE

FACULTY OF SCIENCE AND AGRICULTURE

(School of Mathematical and Computer Sciences)

UNIVERSITY OF LIMPOPO

Supervisor: Professor Mthulisi Velempini

2022

DEDICATION

To my dear family, I dedicate my dissertation.

Mary Ramaboea, my mother.

And to Kganakga Lebepe and Mmathole Ramaboea, my siblings.

DECLARATION

I, ***Mampuele Lebepe,*** hereby declare that this dissertation titled MITIGATION **OF DENIAL-OF-SERVICE ATTACKS IN SOFTWARE-DEFINED-COGNITIVE RADIO NETWORKS USING SOFTWARE-DEFINED-COGNITIVE RADIO SHIELD** submitted at the University of Limpopo in fulfillment of the requirements for Master's degree is my original work and has not been submitted previously to any university or institution of higher learning. I further declare that all sources cited are acknowledged and correctly referenced.

Signature: Lebepe M Date: 12/12/2022

ACKNOWLEDGMENTS

This research would have been difficult to complete without the material and moral assistance of diverse individuals. Therefore, I extend my appreciation to them. Above all, I thank Almighty God for providing me with excellent health and guidance throughout the course**.**

I am very grateful to Professor Velempini for his efficient guidance, commitment, accessibility, and professional guidance. For me, it was by no means an easy path, but because of your availability whenever I needed assistance, the study was smooth. Your criticism, comments, and guidance have helped me tremendously.

I thank North-West University for allowing me to use their NetSim software.

My mother, Mary Ramaboea, thank you for your being an amazing mother. Your love, support, and prayers have done wonders throughout my studies. I appreciate you.

I am grateful to my colleagues from the University of Limpopo for the support that spurred me to complete this study.

ABSTRACT

A ground-breaking approach to managing network resources is provided by software-defined networks that solve a number of management-related concerns. A novel paradigm called Cognitive Radio Networks (CRN) was developed to circumvent spectrum limitations. Methods for efficient dynamic spectrum access are employed. CRN allows secondary unlicensed users to take advantage of the licensed spectrum without interfering with authorized users. Security breaches on these networks are unavoidable. Network security planning is the first step in network defence. This study develops a strategy for software-defined cognitive radio networks that detect and counteract denial of service (DoS) attacks. We develop an intrusion detection system (IDS) to research the consequences of DoS attacks and mitigate DoS attacks in software-defined cognitive radio networks. The IDS is software based and is connected to the software-defined cognitive radio network's controller. We focus on the detection time, or the amount of time it takes to realize an attack has occurred, the payload, or the portion of malware the attacker wants the victim to receive, the jitter, or the variance in the time delay when a signal is transmitted and when it is received over a network connection, and the packet drop rate, or the number of packets lost during an attack. The round trip time and throughput indicate how quickly packets are transmitted during an attack. To generate the findings and compare them to existing schemes, we used NetSim which was installed on the Windows 10 Operating System. We proposed a scheme that detects and mitigates DoS attacks which performed well in terms of the jitter, throughput, detection time, round trip time, and packet drop rate. We compared our scheme to the SDN-Guard. Our scheme achieved less throughput, packet drop rate and round-trip time. Our scheme achieved faster detection time and lower jitter. The payload of our scheme was also less compared to the SDN-Guard.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

CRN -	Cognitive Radio Networks
CUSUM -	Cumulative Sum
DDOS -	Distributed Denial of Service
DL -	Deep Learning
DoS -	Denial of Service
DSA -	Dynamic Spectrum Access
DT -	Decision Tree
IDS -	Intrusion Detection System
ILP -	Integer Linear Program
MLFQ -	Multi-Layer Fair Queuing
PDR -	Packet Delivery Ratio
PSR -	Packet Send Ratio
SDN -	Software Defined Networks
SMTP -	Simple Mail Transfer Protocol
SS –	Signal Strength
TCAM -	Ternary Content-Addressable Memory
WRR -	Weighted Round Robin

CHAPTER 1 – INTRODUCTION

1.1. Introduction

Software Defined Networks (SDN) provides a unique approach to managing network resources that resolve long-standing management issues [1] this approach separates the network's control plane (which determines how data is routed) from the data plane (which forwards the data). Traditional networks combine the control plane and data plane into the same device, making it difficult to administer the network effectively and flexibly. SDN provides enhanced network speed, scalability, and flexibility, as well as simplified management and troubleshooting of a network. SDN enables network managers to design and reconfigure network resources to suit changing demands, monitor and administer the network from a central point by centralizing network control.

SDN is used in a variety of sectors and applications. In cloud computing environments, SDN is employed to provide dynamic and scalable networking architecture. In data centers, SDN is also utilized to enable more efficient and flexible network administration.

Cognitive Radio Networks (CRN) was developed to address the spectrum challenge. CRN is a wireless communication network that employs cognitive radio technology to adapt dynamically to changing communication contexts. Cognitive radio technology enables radios to detect the radio frequency spectrum and make intelligent decisions regarding access and utilization of available frequency bands by utilizing complex algorithms. CRN employs efficient dynamic spectrum access techniques. Secondary unlicensed users can take advantage of licensed spectra without interfering with authorized users.

The radio spectrum that is currently accessible cannot fulfill the requirements of all these wireless devices due to the high cost and broad availability of wireless communication devices. This is the gap that Cognitive Radio fills.

The users of a network are either classified as Primary Users, who can utilize a licensed spectrum that is set aside for their requirements. Secondary Users may use

the reserved spectrum when it is not completely occupied without impairing the performance of Primary Users.

There are several instances of CRNs ranging from military to civilian applications. Military employ CRN technology to communicate in dynamic and unexpected circumstances. CRN technology is used to increase energy economy and network reliability in wireless sensor networks. The Cognitive Radio Sensor Networks (CRSN) project, for example is a CRN-based wireless sensor networks for environmental monitoring. In emergency response scenarios, CRN technology can be utilized to ensure reliable and efficient communication among first responders. These are some of the instances of how CRNs are utilized. We expect more novel uses and applications to emerge as CRN technology evolves.

1.2. Statement of the problem

In wireless networks, the DoS attack is a recurring problem. A DoS attack is a network attack that aims to stop the server from providing its clients with services [2]. DoS attacks are using more and more complex attack tactics that overwhelm networks and make resources unavailable. DoS attacks can stop services from working, costing money. Although some companies use reverse proxies as reliable barriers against DoS attacks, these attacks still occur. Therefore, there is a need for more powerful and efficient DoS attack mitigation strategies [3] [4].

In this study, we use the SD-CRN Shield to reduce DoS attacks that occur in software-defined cognitive radio networks (SD-CRN). Merging SDN and CRN is known as SD-CRN. SD-CRN are wireless communication networks that employ cognitive radio technology to adapt to changing communication environments. Cognitive radio technology allows radios to detect the radio frequency spectrum and make intelligent decisions about how to access and utilise available frequency bands using SDN approaches.

The application of cognitive radio technology in SD-CRN enables radios to adapt dynamically to changing network conditions and maximize performance. Nevertheless, this flexibility exposes its weaknesses to attackers who can lauch DoS

attacks. For example, an attacker may send a high number of queries to a radio, forcing it to switch frequency channels repeatedly and waste a substantial amount of resources while preventing other legitimate users from accessing the network.

Furthermore, in SD-CRN, the centralized controller is in charge of network management and coordination, making it a vital component. As a result, if the controller is attacked by a DoS attack, the entire network may become unavailable.

1.3. Aim

The study aims to design and evaluate the DoS attack mitigation scheme in SD-CRN.

1.4. Objectives

The objectives of the study are intended to:

i. Design an effective IDS to counter the effects of DoS.

ii. Investigate the impact of the placement of the IDS.

iii. Evaluate the effectiveness of the proposed scheme.

1.5. Hypothesis

The proposed scheme for DoS attacks in SD-CRN is intended to enhance the detection rate and mitigate DoS attacks. The IDS implementation is expected to improve the network's capacity, identify and mitigate DoS attacks, resulting in a more secure and resilient SD-CRN architecture.

1.6. Research questions

i. What is an effective way of designing a DoS countermeasure?

ii. How effective is IDS placement in addressing the DoS attacks?

iii. To what extent can the proposed SD-CRN Shield achieve robust results?

1.7. Rationale

As SD-CRN become more common, there is an increasing need to address the security concerns. Recent research has demonstrated that SD-CRNs are vulnerable to DoS attacks, which can be catastrophic. As a result, the purpose of this study is to evaluate relevant studies on the consequences of DoS assaults on SD-CRNs and propose a secure scheme.

The significance of security vulnerabilities in new technology is what inspired this study. The DoS attack interferes with networks, making them unavailable or inefficient. The attacks may result in business or revenue losses for organizations. Network security against malicious users makes services and networks more accessible, ensuring continued business operations.

To overcome the security issues with SD-CRN technologies, there is increased interest in their use. We assess the relevant research that aims to address the DoS effects on SD-CRN. We review earlier work and consider how it relates to our own. The theories and methods that guided the selection of the study's approach and its research topic are discussed subsequently.

1.8. Scientific contribution

This study implements a DoS detection and mitigation scheme in SD-CRN. It also investigates the effectiveness of the SD-CRN Shield in reducing the effects of denial of service attacks in the software-defined cognitive radio environment. The study enhances and consolidates the security of emerging technologies and networks. The research study adds to the body of knowledge in SD-CRN security.

1.9. Ethical considerations

The study does not require ethical clearance.

1.10. Availability of research infrastructure

Resources are available from open-access data and tools at the University of Limpopo to facilitate the full investigation and resolution of the research problem.

1.11. Dissertation outline

This dissertation consists of five chapters. Chapter two presents the literature review. We discuss the overview of SD-CRN and review the literature on DoS attacks and their impact on SD-CRN. We provide a summary and critique of existing research on SD-CRN and DOS attacks. Chapter three presents the methodology. We discuss the research design, approach, methods and techniques used. Chapter four analyzes the results. We present and analyze our results. We evaluate the efficiency of IDS placement in addressing the DoS attacks in SD-CRN. We assess the proposed SD-CRN Shield and its capacity to generate reliable results. We then explain the research implications and relevance. Chapter five concludes our work. We conclude our study by making a final remarks based on our findings. We provide a summary of the research objectives and hypotheses and an overview of the research methodology and results. We also discuss recommendations for future research.

CHAPTER 2 - LITERATURE REVIEW

2.1. Introduction

The need to address SD-CRN software's security issues is expanding along with its adoption. SD-CRN have grown in popularity due to their capacity to learn from their

surroundings, adapt to various communication settings, and dynamically distribute resources, Yet, because of their expanding use, SD-CRN has become a prominent target for cyber-attacks. DoS attacks, in particular, have become a serious worry for SD-CRN.

To overcome these issues, numerous ways to mitigate DoS attacks in SD-CRN have been proposed. These methods vary from reactive measures like traffic filtering and rate restriction to more proactive ones like intrusion detection and prevention systems. The efficiency of these measures, however, varies according to the resource availability, network structure, and attack scenario.

This chapter reviews related research that assesses the DoS in SD-CRN. It evaluates prior findings and establishes connections to our current study. It explains the concepts and methods supporting the decision to use a certain research focus and approach.

The articles used in this chapter offer a wide range of research projects, including simulation-based experiments and empirical evaluations. These were chosen to offer a thorough overview of the many facets of DoS attacks in SD-CRN, including attack types, detection and mitigation techniques, and efficacy. Overall, the publications included in this review were chosen to give a fair and comprehensive examination of the issue of DoS in SD-CRN, highlighting the difficulties, solutions, and future research objectives in this field.

2.2. Related works

Attacks that prevent genuine users from using a server's service are among the most severe threats. One of the most dangerous threats is a DoS attack, which aims to prevent genuine users from using a server's service [5]. There are two groups of these attacks. The first type is referred to as flooding-based attacks. Here, the attacker bombards the target with numerous packets, depleting its resources. Vulnerability attacks are the second type. These are distinguished by sending a message that has been specially written to the targeted hosts and denies the service being offered [6].

DoS assaults have moved to a new class in which both flooding, and vulnerability attacks are integrated as technology advances. It is difficult to distinguish between malicious and valid communications in this category [7].

In [8], the authors developed a method for minimizing the consequences of DoS attacks in SDN by installing parallel flows. While the majority of current methods for handling DoS attacks in SDN typically reject harmful packets or aggregate flow rules, this results in the deletion of valid packets or a loss of good control across network traffic. Their approach results in a significant reduction in control channel traffic and controller usage by installing flow rules on a single request from the source throughout all switches along the path from the source to the destination. By contrasting the approach with the crucial SDN controller, the approach is evaluated. The technique increased SDN efficiency concerning reaction time, CPU utilization, and bandwidth.

Authors in [9] performed a safety analysis of a STRIDE model and offered recommendations for preventing DoS attacks in SDN. Rate limitation, flow timeout modifications, and flow aggregation were suggested to prevent DoS attacks in SDN.

It was suggested that jamming attacks could be detected in cognitive radio networks [10]. During a jamming attack, attackers are either external users or secondary users. An IDS is crucial because cognitive radio networks are sensitive to security attacks. To identify odd behavior in CRN, the cumulative sum (CUSUM) was used. To counter security risks to cognitive radio networks, countermeasures are implemented. An IDS offers new ways of reducing attacks on the architecture of cognitive radio networks.

To counter DoS attacks, the Multi-Layer Fair Queuing (MLFQ) approach was suggested [11]. By keeping Packet-In message queues in the controller, the approach minimizes DoS threats. The controller pools requests from various queues using the Weighted Round Robin (WRR) algorithm. A queue expands into a per-switch queue when the amount of packets present in the queue reaches a predetermined edge.

A queue is enlarged into a per-port queue if it is still larger than it was before. The queues are once again combined into one queue, however, if the size of the sub-queue falls below the set threshold. The hosts that are legitimate and are connected

to the switch will experience additional lag whenever the attacker is present due to the increased computation required by this method. The queues are once again gathered into one queue if the size of the sub-queue dips beneath the boundary. With this technology, multi-layer queue boards need to be calculated more thoroughly, which means that when an attacker is present, the switch's actual actions will be delayed more.

The Flow-Ranger system, which was suggested by authors in [12], enables the detection and mitigation of DoS threats. It is executed on the side of the controller and has three parts: a trust management part that assigns a trust value to each packet-in message based on its source; a queuing management part that gives the message priority based on its level of trust; and a message scheduling part that handles messages using a weighted Round Robin algorithm. By ensuring that genuine flows are served first in the controller, Flow-Ranger reduces the effects of DoS attacks on the network performance.

Distributed Denial of Service (DDoS) attacks frequently target the SDN controller due to its high vulnerability [13] [14]. Although the SDN controller is extremely exposed and frequently a target of DDoS attacks, the authors in [15] argue that this presents a significant chance to lessen DDoS attacks within cloud computing systems.

Flood-Guard is a proposal that was put forth by [16]. It protects against DoS attacks known as "flow request flooding" attacks. When the controller notices a DoS attack, the scheme introduces a rule placed at the attacked switch to reroute all the new flows to a data location cache. This Flood Guard drawback is that complicated controller applications may not always utilize all available execution paths, necessitating the placement of additional devices on the data plane.

In [17], the authors suggested Avant-Guard. Whenever the controller is unprotected, TCP SYN flooding can overwhelm its resources, causing the CPU, memory, and bandwidth to be used up by flooded requests. Switches can delegate all TCP connections using Avant-Guard. The Avant-Guard will submit a flow request following the completion of the handshake for a Transmission Control Protocol (TCP)

connection. The fundamental flaw in this technique is that it can only prevent SYN flooding.

The authors of [18] developed a performance and security analysis approach. This model determines the ideal encryption key length for the network's best performance and security. The expected encryption times when the network performs at its peak are measured by the model. As seen in Figure 1, it has an intrusion detection system. However, the model was unable to identify DoS attacks in SD-CRN and could only identify assaults that intercept and alter the content.

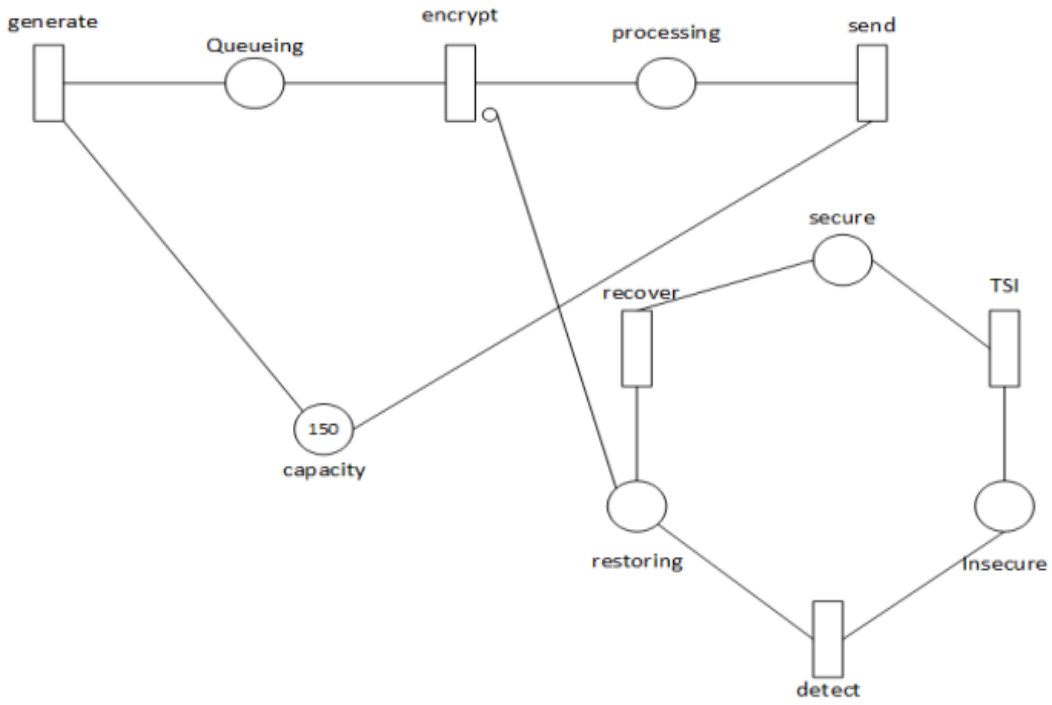

Figure 1: Petri net model for combined performance and security analysis

In an SDN, an SRL architecture was suggested to mitigate TCP SYN-flooding [19]. Two modules from the SRL that are implemented in the controller help to minimize the attack. There are two modules, a flow module, and a hashing module. TCP SYN Flood attacks were conducted once the structure was integrated into a floodlight controller. The hash module determines the hash values using the IP address that are from the SYN flood packets. After that, the flow rules are then stored by the controller using their hash value. Those with a low hash value are eliminated since they are thought to

be attacks. Wildcard rules are computed using faked IP and MAC addresses by the flow module. Then, all the requests coming from the bogus IP addresses are blocked.

A Flow-IDS system, seen in Figure 2, was presented by [20] to use SDN for discovering and reducing Simple Mail Transfer Protocol (SMTP) Flood attacks, a sort of DoS attack. The scheme is used to identify SMTP communication flows that are malicious. To determine if the flows are legitimate or not, calculations and decisions based upon Deep Learning (DL) and Decision Tree (DT) algorithms were made. The authors demonstrated that SMTP attacks on software-defined networks can be stopped and detected using Flow-IDS.

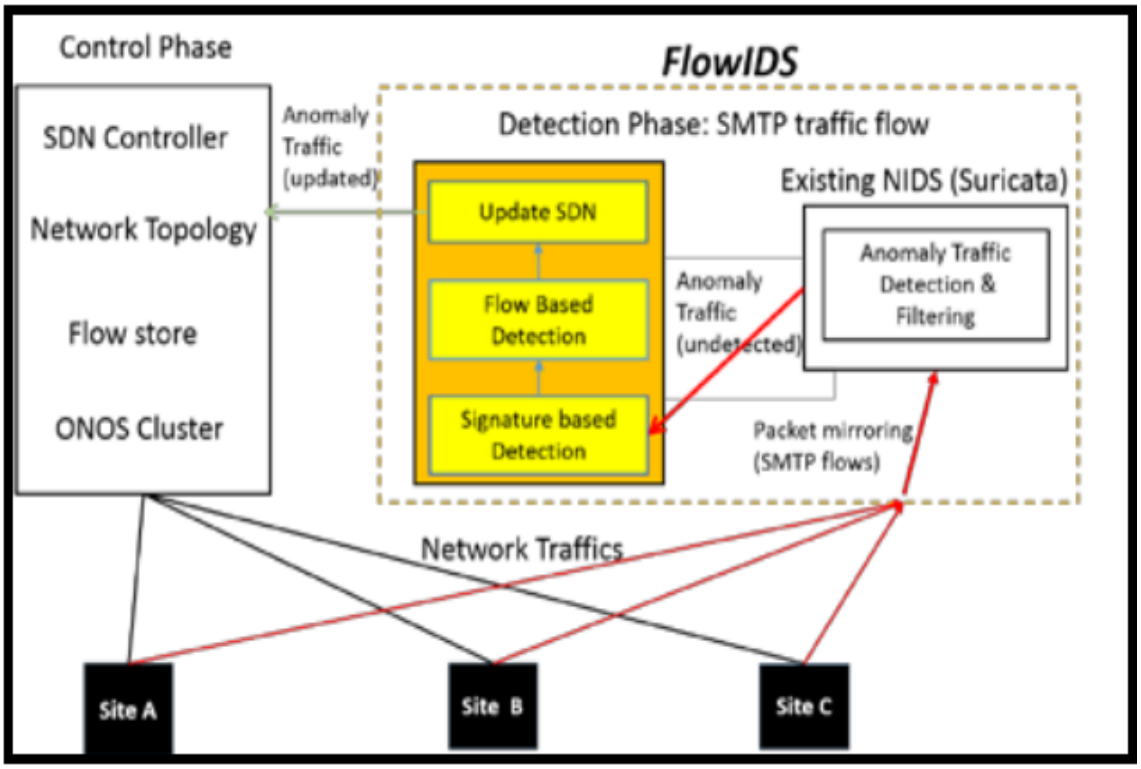

Figure 2: Flow-IDS Framework

To minimize the impact of DoS in SDN, Shin [21] used a strategy utilizing parallel flow implementation. The plan was successful in reducing the DoS attack without compromising network administration and monitoring capabilities. On one request coming from the source, the switches that are in the source-to-destination path are configured to install flow rules. This lowers the volume of traffic on the control channel

and the controller's utilization. The CPU utilization, flow requests, control channel bandwidth, and response time of the SDN were all enhanced.

Authors in [22] proposed SDN Manager. It has up to five parts that cooperate to quickly minimize DoS attacks on SDN. A storage device, an updater, a checker, a monitor, and a forecast engine are some of the constituent parts. The scheme uses a loop to read the statistics of the flow. The network is updated after anticipating the changes in the flow of the bandwidth based on collected data. Results demonstrated that the plan only marginally enhanced the SDN infrastructure.

In cognitive radio networks, it might be challenging to distinguish between networks facing a DoS assault and networks that are congested [23]. In order to address this, queue length, packet loss probability, and throughput are calculated in conjunction with the packet send ratio (PSR) and packet delivery ratio (PDR) to acquire the results and then to distinguish between the two. A program known as SDN-Guard was designed by authors in [2]. In this approach, the workload usage of average processing time, network bandwidth, controller load, switch-to-controller bandwidth, and unusual node detection could all be reduced.

DoS threats and associated solutions in SD-CRN were examined in [24]. The authors outlined many forms of DoS attacks that may be conducted against SD-CRNs, such as jamming, resource depletion and flooding attacks. They also examined many solutions proposed to counteract these threats, such as intrusion detection and prevention systems, traffic filtering and routing protocols. The article identifies research gaps and offers future research initiatives to solve SD-CRN vulnerabilities to DoS attacks.

Authors in [25] proposed an effective DoS attack mitigation technique for SD-CRN based on SDN and blockchain. SDN is used to manage network traffic and traffic flow, while blockchain is used to provide a secure and dependable control plane for SDN. The authors run simulations to evaluate the proposed scheme's performance under various attack scenarios, and the findings suggest that the proposed scheme may reduce DoS attacks in SD-CRNs with little overhead and latency.

The author in [26] designed and implemented a security guard model to solve DDoS attacks on POX SDN controllers. The model, named SGuard, is represented as a novel five-tuple feature vector utilized for classifying traffic flow by employing SVM. Mininet was used to evaluate SGuard in network and software environments. The SGaurd was evaluated in terms of system's performance in terms of accuracy, traffic flow, bandwidth, and delay.

DoS attacks continue to pose a threat to the internet because they advance more quickly than technology and are challenging to defend or avoid. None of the above approaches could simultaneously focus on the IDS placement and signal strength because none of them were applied in an SD-CRN context.

2.3. Summary

Wireless networks are still at risk from DoS attacks. Unfortunately, DoS attacks are keeping up with technological innovation. It is getting harder to detect and deal with DoS attacks as wireless networks become more advanced. The Jamming Attack Defender [10] and the SDN-Guard [2] are the top-performing strategies for reducing the DoS in SD-CRN, according to a review of the literature and the findings of this study. To provide encouranging results, the two strategies are integrated and modified.

CHAPTER 3 – METHODOLOGY

3.1. Introduction

This chapter is aimed at offering a clear and complete description of the approaches utilized to address the DoS attack problem in SDCRNs. It describes the general strategy and structure used to develop and implement the scheme, the design of mitigation mechanisms, and the efficacy evaluation.

The methodology section of this study will discuss the procedures used to build and test a technique for mitigating DoS attacks in SD-CRN. It will lay out the research strategy, and the development of mitigation strategies based on existing research. It will also describe the processes used to test the scheme's efficiency using simulation-based experiments, as well as the analysis of the findings acquired.

In our study, we design and assess the software-defined cognitive radio shield (SD-CRN Shield) in SD-CRN, a DoS attack mitigation method. Therefore, the proposed strategy is designed, put into action, and its effectiveness is assessed in the next chapter. The platform used and the parameters are shown in Tables 1 and 2, respectively.

Table 1: Platform for simulation

Computer	DELL Vostro 3591
RAM	12,00 GB
CPU	Intel(R) Core (TM) i5-1035G1 CPU @ 1.00GHz 1.19 GHz
OS	Microsoft Windows 10 Pro

Table 2: Parameters and tools

Parameters Used	Tools Used
Network Simulator	NetSim
Simulation Area	500m*500m

NetSim is used to emulate the network topology. It builds a network of virtual hosts, connections, controllers, and switches. NetSim provides a number of techniques designed to collect data during the simulation, including logging features, and built-in monitoring capabilities where these tools allow users to monitor many network parameters such as jitter, throughput, delay, and packet loss. NetSim also supports external data collecting and scripts. This data may be used to study the simulated network's performance, identify possible flaws or areas for development, and assess the efficacy of various network setups and protocols. NetSim also allows users to create customized reports that incorporate specific metrics and factors of interest. These reports may be exported to CSV, Excel, and PDF file formats.

DoS attacks attempt to flood a network or server with traffic or requests, rendering it inaccessible to genuine users. The larger the network, the more resources it has to manage including high traffic volume, making successful DoS attacks more difficult to detect.

Larger networks, on the other hand, may be more appealing targets to attackers because they contain more valuable data or resources to attack or disrupt. Moreover, bigger networks may have more access points, making them more vulnerable to attacks.

The impact of a DoS attack on a network is generally determined by a number of factors, including the type and severity of the attack, the network's security measures, and the network's ability to manage traffic. Strong security procedures, as well as enough resources and redundancy, can assist in reducing the impact of DoS attacks on any network size.

3.2. SD-CRN Shield

SD-CRN Shield was proposed in this work. It is plugged into an SDN controller. In order to inspect the traffic of the network and produce warnings when malicious traffic is identified, it uses an IDS.

The SD-CRN Shield is derived mainly from the structure of the SDN-Guard and the Jamming Attack Defender. The scheme is made up of two units. The first one is a flow management unit, which chooses the paths for routing the flows and establishes the hard timeout of the correlating Ternary Content-Addressable Memory (TCAM) entries. This is done based on the likelihood that the flow poses a threat and flows are managed to reduce the impact of DoS assaults. To decrease the number of entries utilized in the switches Ternary Content-Addressable Memory, a rule aggregation module is put in charge of collecting flow entries of harmful traffic. To alert the SD-CRN Shield of any anomalies, an IDS is developed and shall be in continual communication with it.

The suggested scheme is based on the following three design considerations to mitigate the DoS attacks:

i. Timeout management: Using the IDS, determine the timeout value for each flow rule based on the likelihood of a threat.
ii. Aggregation of malicious flow rules: Hard timeouts are given to malicious flows. They will be present therefore in the TCAM tables for a considerable amount of time. As a result, they will be overloaded with more used entries. In order to fix this, malicious flow entries are merged if they have the same source, destination, and link going out.
iii. Threat based routing: This method routes harmful data through the network's least-used channels to conserve bandwidth.

In networks that use DSR at Layer 3, the SD-CRN Shield has a code for avoiding the malicious node and choosing perhaps another path. The function validates the route

reply in the route cache and searches for the malicious node when the node is a malicious node and a route reply is processed.

When a malicious node is identified, its route entry is deleted from the cache. Code for the IDS functioning at Layer 2 is also included in the scheme. A watchdog timer starts when a packet is sent if _NETSIM WATCHDOG_ is defined. After a packet has been forwarded to the next hop node, the current node checks the watchdog timer duration to decide whether to keep sending it on to the target node.

The malicious node does not forward incoming packets. The malicious node's watchdog timer runs out in the node that transmitted the packet to it. There is a counter that records each time the watchdog timer expires. This is the number of packets that are sent out but are not forwarded by the next hop node. The current node flags the following node as a malicious one when the value of this counter crosses the failure threshold.

The measures below are the main focus of this study:

i. Packet Drop Rate
ii. Jitter
iii. Round trip time
iv. Throughput
v. Payload
vi. Detection time

3.3. IDS placement

The strategic positioning of the IDS inside a network architecture to identify and respond to security threats is referred to as IDS placement. The location of IDS is crucial to the overall efficiency of an organization's security architecture. IDS systems that are correctly placed and configured can assist in detecting and preventing security attacks while reducing the risk of data loss, network outage, and reputational harm.

We examine and determine where the IDS should be placed. We do this to determine whether or not the location of the IDS is significant. We first deploy a number of IDSs, each of which is attached to every switch. Each IDS examines the network traffic that passes through the switch that it is connected to. We then set up one IDS that monitors all network traffic.

To represent the IDS placement problem, we suggest using an integer linear program (ILP).

We Let $G = (N, L)$ be a graph that represents the network

N: number of switches

L: links that connect to the switches

Cost of shortest path from switch $n \in N$ is defined as $p_{n\overline{n}}$

To denote a flow crossing the network, Let $i \in I$.

The flow's throughput is denoted f_i of the flow i.

Define $r_{in} \in \{0,1\}$ as a Boolean that equals to 1 if the flow $i \in I$ crosses the switch $n \in N$

To determine if flow i is mirrored from switch n to the IDS, we define a decion variable $x_{in} \in \{0,1\}$ as a Boolean. Evey flow i is only mirrored to the IDS once. Therefore, we must adhere to the following restriction:

$$\sum_{n \in N} x_{in} = 1 \qquad \forall\, n \in I. \tag{1}$$

A flow i cannot be forwarded from a switch n if it does not pass this switch, therefore the variables defined thus far are known to the controller. As a result, we have:

$$x_{in} \leq r_{in}\ \forall n \in N\ \forall i \in I. \tag{2}$$

The amount of mirrored traffic transferred from the switches to the IDS corresponds to the cost of mirroring all flows to an IDS attached to a switch $\overline{n} \in N$. We calculate it as:

$$the\ C_{\overline{n}} = \sum_{i \in I} \sum_{n \in N} x_{in} p_{n\overline{n}} f_i \qquad \forall\, n \in N. \tag{3}$$

Finding the switch $\overline{n} \in N$ that reduces the cost of mirroring is our ultimate goal:

$$\min_{\overline{n} \in \overline{N}} C_{\overline{n}} \tag{4}$$

The location of the IDS (ie, $\overline{n}$) and the switches that should transfer traffic to the IDS (i.e., using the decision variable x_{in}) are both provided by the proposed integer linear program (ILP).

3.4. Packet Drop Rate

By observing the packet drop rate (PDR) and signal strength (SS), a DoS attack can be detected. The PDR of a user is calculated as the ratio of packets transmitted to packets received by the user. If an attack is conducted against a secondary or unauthorized user, the SS is investigated. If the SS is too high, the packet is dropped. To find changes in the PDR of the secondary user, the CUSUM algorithm shall be used. A warning that a DoS attack is in progress will be given if the calculated mean returns a number that is higher than the predetermined value and the SS at the secondary user is high. We shall suppose that the random sequence's mean value is negative and that it turns positive if any modifications are found [10].

One way to determine the PDR is:

For $q(q \leq n)$

$$PDR = \frac{q!}{n!(n-q)!} * p^q(1-p)^{n-q}$$

where n = number of trials.

q = number of successes

p = probability of success

The probability of success can be calculated by dividing the number of successful outcomes by the total number of possible outcomes.

Each packet contains the SS, which is taken from the physical layer.

3.5. Throughput

To examine the behaviour of controllers, we evaluate the source-to-destination possible throughput as well as the incoming controller throughput.

3.6. Jitter

When assessing the variability in ping, jitter is the variation in the time delay between when a signal is delivered and when it is received via a network connection. Jitter is used to assess the network's performance. It is when there is a time delay in the sending of these data packets over one's network connection. We measure the jitter level by calculating the average time difference between each packet.

3.7. Payload

Payload is the part of transmitted data in the actual intended message. In the context of a DoS attack, the payload is the portion of malware that the attacker intends to deliver to the victim.

3.8. Detection time

Detection time is the time it takes to identify that an attack has occurred. This metric is important because the quicker an attack is detected, the quicker it can be mitigated.

3.9. Summary

The study's methodology assists in achieving the study's aim and objectives. It also aids in answering the research questions of the study. The tools, algorithms, and equations applied were carefully selected. The simulations generated sufficient data to provide comparative results of the schemes and to effectively evaluate their performance.

CHAPTER 4 – RESULTS AND DISCUSSIONS

4.1. Introduction

The most common and consistent threat to SD-CRN networks is DoS. The attack aggressively overloads the targeted servers and network links with malicious traffic until they are unable to service legitimate users.

To mitigate the risks associated with DoS attacks, we proposed a scheme called Software Defined-Cognitive Radio Shield. We used a hybrid of cognitive radio and software-defined networking approaches to construct an intelligent network shield capable of detecting and mitigating DoS attacks in real time.

We evaluated the effectiveness of our proposed scheme using five key metrics, namely, detection time, PDR, RTT, jitter, payload, and throughput. The results of our experiments are presented in this chapter.

4.2. IDS Placement

Using sampled traffic, we assessed the IDS's accuracy in the first section of our analysis. We examined how quickly packets were processed at various sampling rates. In our experiment, we generated a TCP-SYN flooding DoS attack and conducted a number of trials with various sampling rates. The percentage of detected attacks was calculated by dividing the number of attacks successfully identified when sampled traffic was evaluated by the total number of attacks discovered when all traffic was analyzed.

The best possible location for the IDS is one that minimizes traffic to the IDS location and minimizes bandwidth consumption by the traffic. We used 8 switches for our research, thus we first installed the IDS at the controller before placing it on each of the 8 other switches. The results demonstrate that the controller, as depicted in Figure 3, is the ideal location for the IDS.

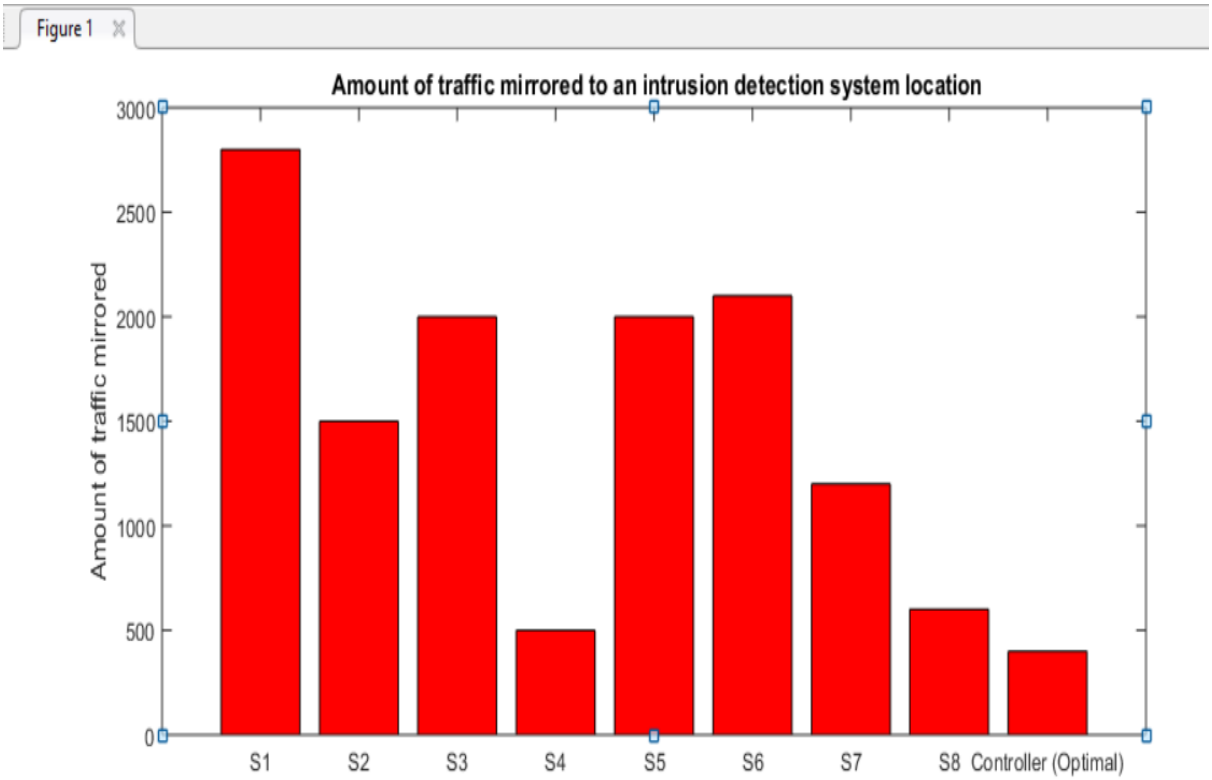

Figure 3: Amount of traffic mirrored to the IDS location.

Comparing the controller to switches 1 through 8, the controller has the least amount of mirrored traffic. This shows that the controller is the ideal location. S4 has the second least amount of mirrored traffic but it is not the lowest, so we take the lowest for the best place to locate the IDS. Five simulations were run for to determine the optimal location for the IDS.

4.3. Simulation Scenarios

We created 5 scenarios to evaluate the performance of our scheme as shown in figures 4 to 8. The network scenarios consist of two Wired Nodes, one L2 Switch, two routers, one Access Point, one wireless node in the grid environment and the malicious nodes. Traffic is generated from the Wired node to the Wireless node. Figure 4 shows an analysis of the network without malicious nodes. Figure 5 shows the network with one malicious node and Figure 6 shows the network with two malicious nodes. Figure 7 shows the network with five malicious nodes while Figure 8 shows the network with ten malicious nodes.

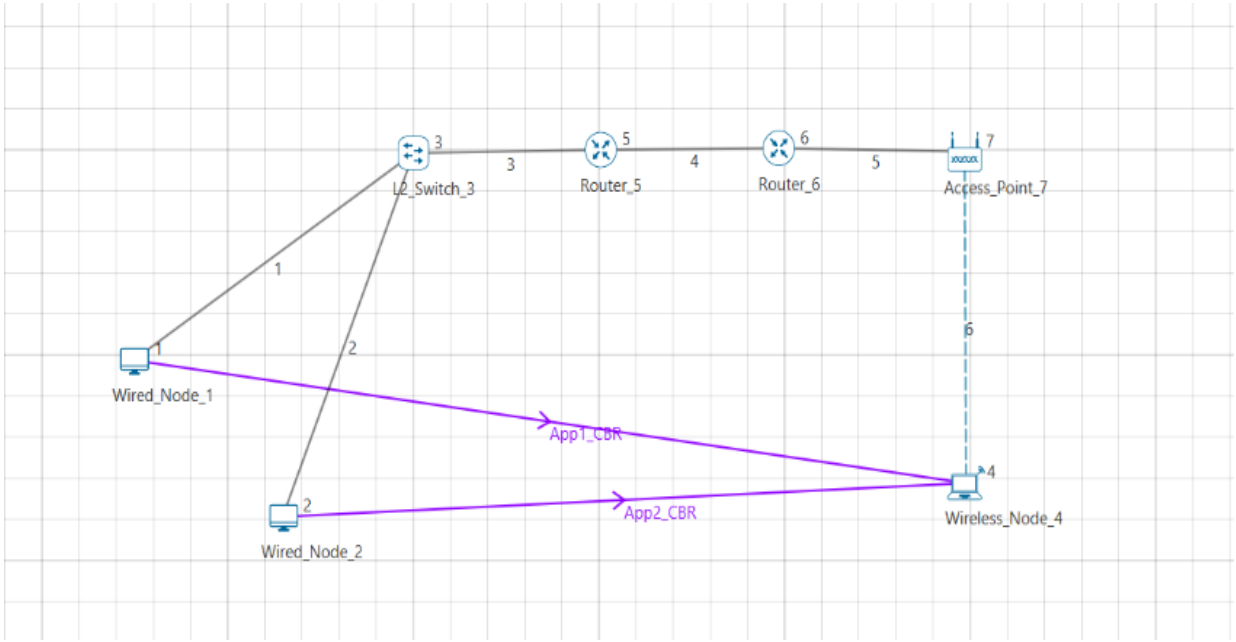

Figure 4: Scenario 1 - Analysis of a Network without Malicious Nodes

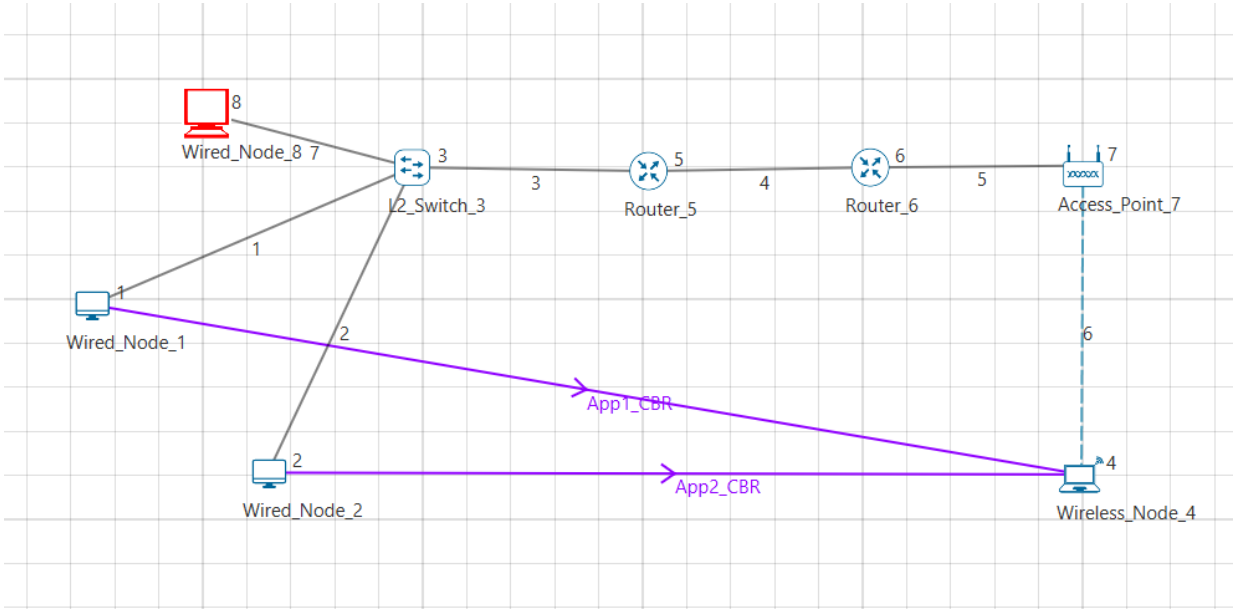

Figure 5: Scenario 2 - A Network with one Malicious Node

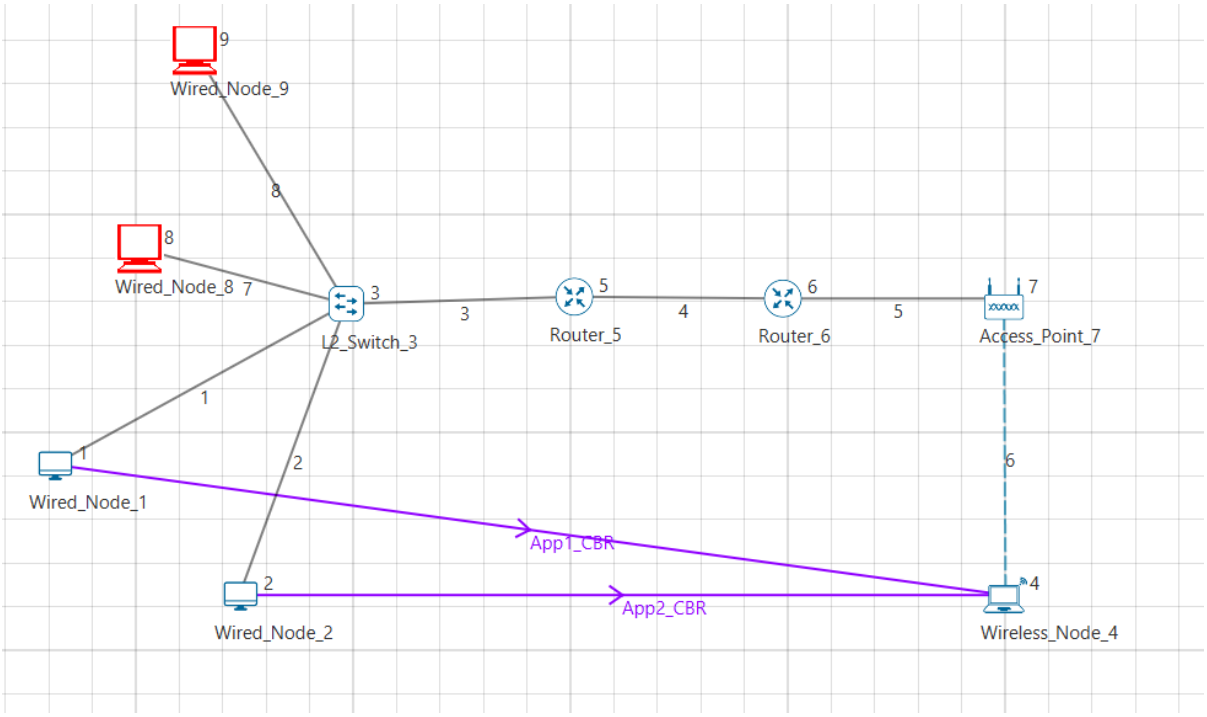

Figure 6: Scenario 3 - A Network with two Malicious Nodes

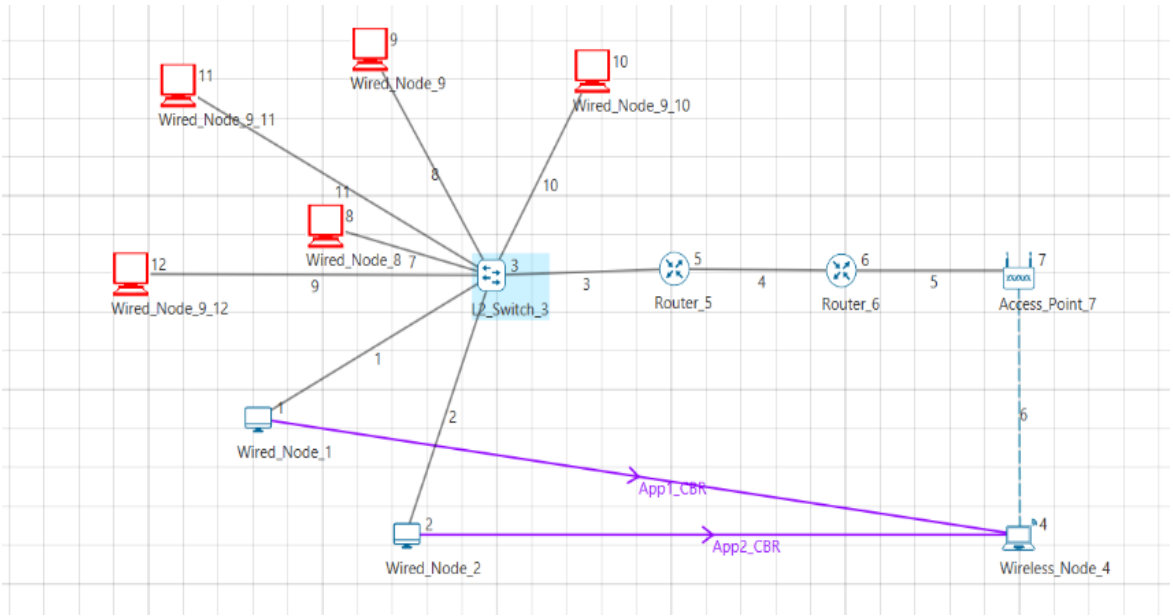

Figure 7: Scenario 4 - A Network with five Malicious Nodes

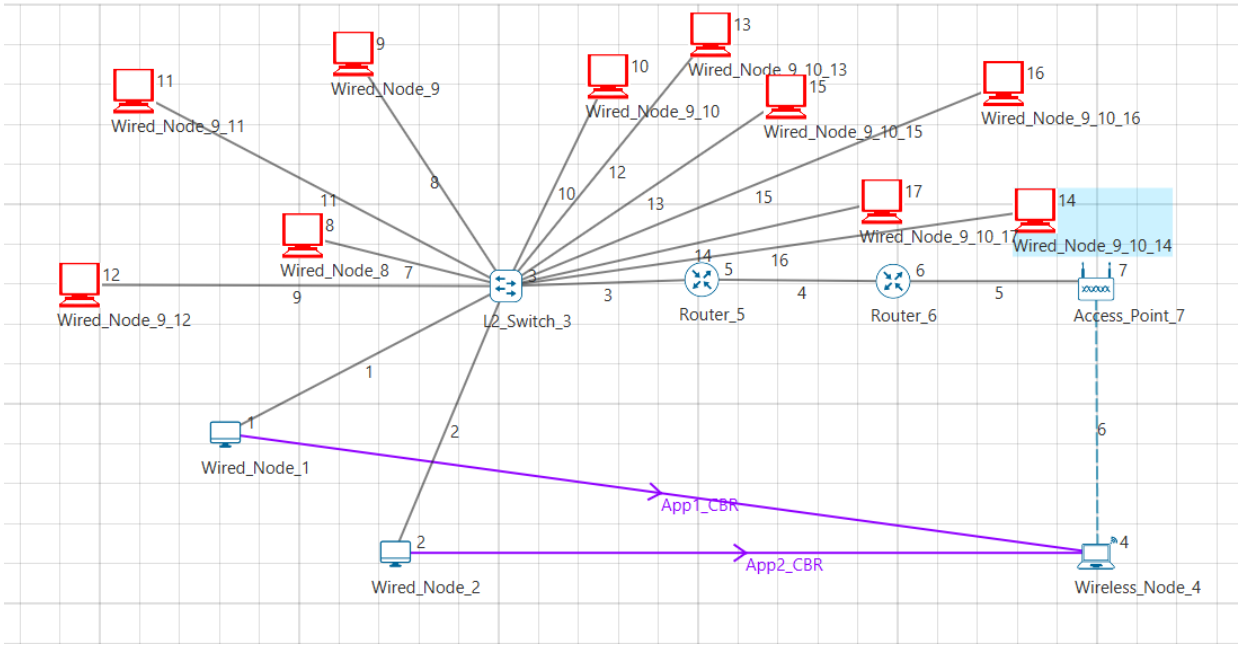

Figure 8: Scenario 5 - A Network with ten Malicious Nodes

4.4. Throughput

We evaluated the two schemes of the throughput. Source-to-destination throughput and controller throughput was measured to analyze the performance of controllers.

Table 3: Throughput scenarios

	SDN-GUARD (MBPS)	SD-CRN (MBPS)	SHIELD
SCENARIO 1: 0 MALICIOUS NODES	0.520928	0.520928	
SCENARIO 2: 1 MALICIOUS NODE	0.810496	0.578160	
SCENARIO 3: 2 MALICIOUS NODES	0.317328	0.286160	
SCENARIO 4: 5 MALICIOUS NODES	0.254457	0.122640	
SCENARIO 5: 10 MALICIOUS NODES	0.094385	0.061904	

Given the five scenarios (figures 4 to 8 and table 3) we considered, we observed the throughput of the malicious nodes decreases for both applications as we increase the malicious nodes because of the SYN flood from the malicious nodes. In scenario 1, there is no malicious node, so the throughput remained the same. For scenario 2, we had one malicious node. The results show that the SD-CRN Shield was able to decrease the throughput from 0.810496 to 0.578160. For scenario 3 we had two malicious nodes and the achievable throughput further decreased from 0.317328 to 0.286160. In scenario 4, the throughput decreased from 0.254457 to 0.122640. This pattern was observed in scenario 5 in which the throughput decreased from 0.094385 to 0.061904. The throughput results are shown in Figure 9.

4.5. Throughput analysis

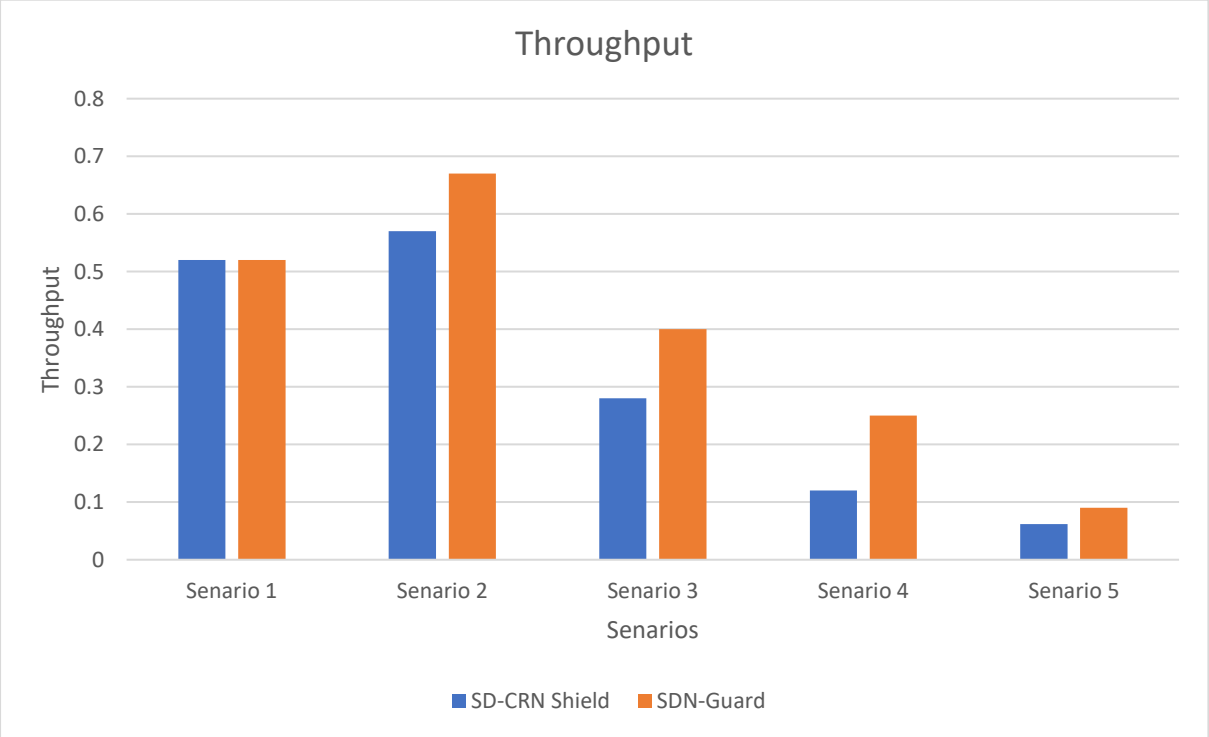

Figure 9: Throughput analysis

The results show that the throughput for scenarios 2 – 5 of the SDN-Guard is more than that of the SD-CRN Shield. Therefore, during a DoS attack, there is a flow in the packets received by the controller. However, it is clear that the SD-CRN Shield significantly lowers the malicious node's throughput. The SD-CRN Shield reduces the requirement to ask the controller for new flow rules by setting high hard timeouts on the malicious traffic. There is a decrease in the number of packets lost in the SD-CRN Shield because the malicious traffic is balanced across the least utilized links which minimizes congestion.

The results from the 5 scenarios illustrate that the proposed scheme was able to reduce the throughput of the malicious nodes. Therefore, this means that the SD-CRN Shield is efficient in reducing the throughput. Not only does the scheme detect the DoS attack, but it also mitigates it and reduces the throughput.

4.6. Packet Drop Rate

PDR is the ratio of the total number of sent packets to the number of packets that were not received. We investigated this to verify how many packets were lost when the DoS attack is launched and measure how many packets our scheme could prevent from being lost when the DoS attack is launched.

Figure 10 presents the PDR when the DoS attack is launched with five malicious nodes and without the SD-CRN Scheme activated.

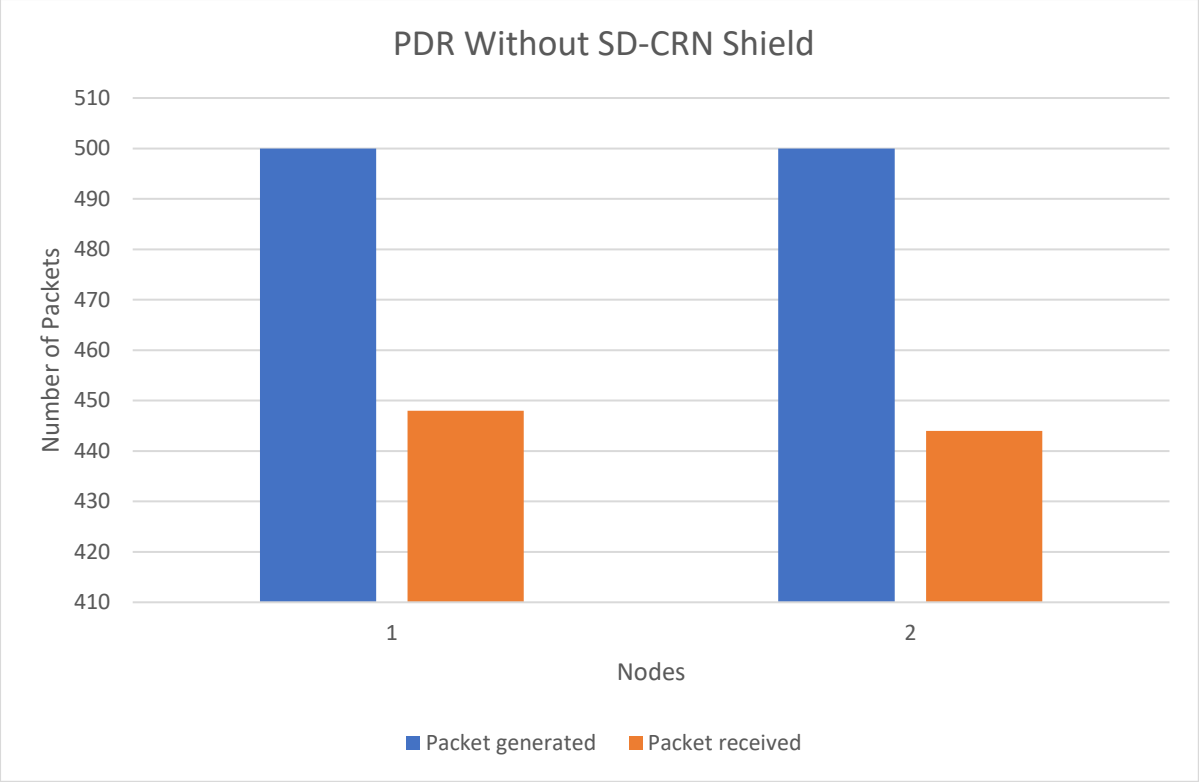

Figure 10: PDR Without SD-CRN Shield

We observed that many packets were dropped when the DoS attack is launched as shown in Figure 10. Node 1 and node 2 generated 500 packets each but received 448 packets and 444 packets respectively. This means that 11.2% of the packets were dropped. Figure 11 shows the results when the SD-CRN Shield is implemented.

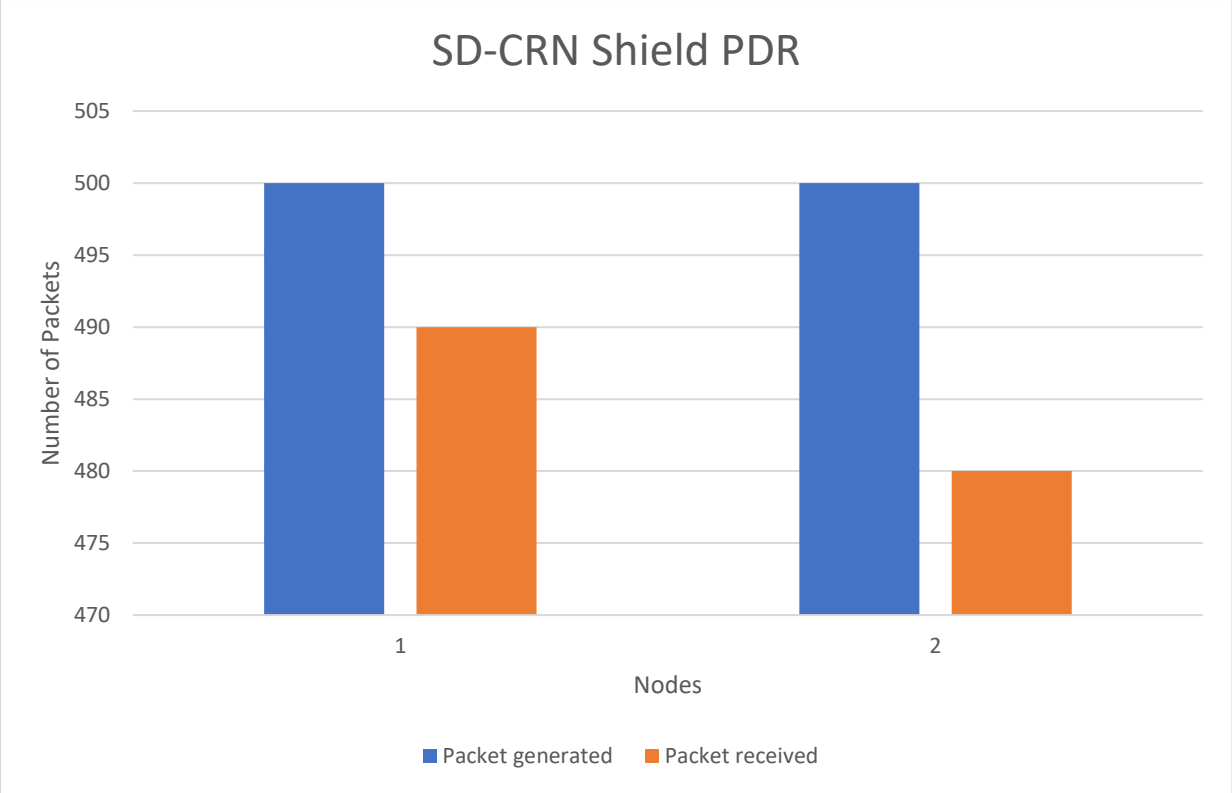

Figure 11: PDR with SD-CRN Shield

The results in Figure 11 demonstrate that fewer packets were dropped when the SD-CRN Shield is implemented. Node 1 and node 2 generated 500 packets and it can be observed that Node 1 received 490 packets while Node 2 received 480 packets. This means that 2% of the packets were dropped for node 1 and 4% of the packets were dropped for node 2. Our scheme was able to reduce the number of packets that were dropped during an attack. This shows that the scheme is efficient in reducing the number of dropped packets during the attack. We then compared the performance of SD-CRN Shield to the SDN-Guard using the PDR. Node 1 sends packets to Node 4 through the App1_CBR as shown in figures 4 - 8 and receives packets from the switch. Node 2 sends packets to Node 4 through the App2_CBR as shown in figures 4 - 8 and receives packets from the switch.

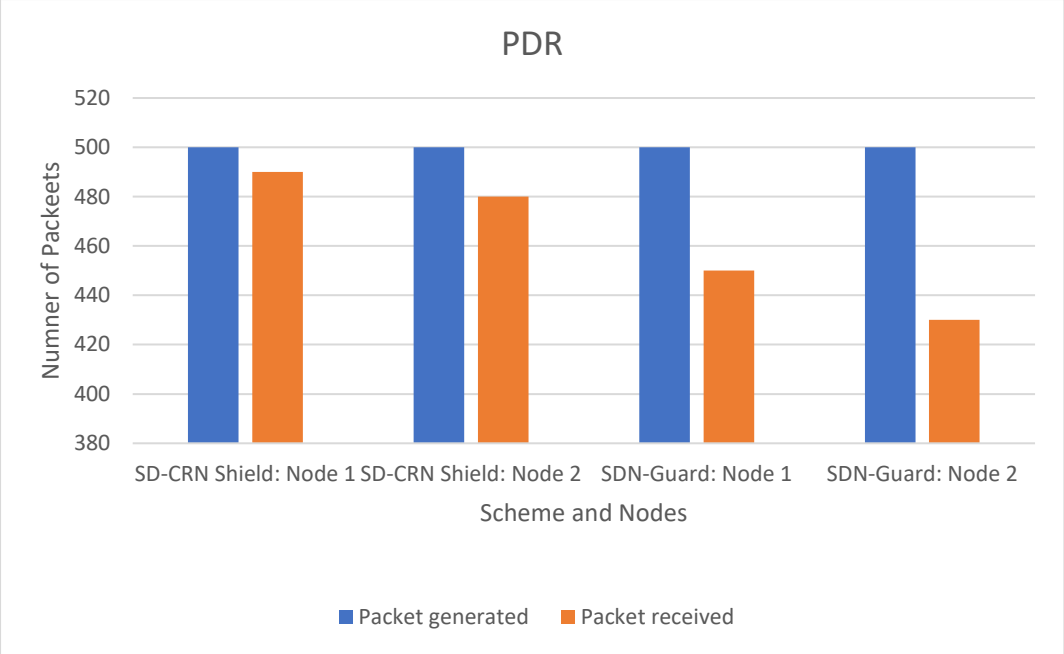

Figure 12: Comparative PDR Results

In the SDN-Guard, less packets were received compared to the SD-CRN Shield as shown in Figure 12. The same number of packets were generated for the two schemes. SD-CRN Shield has a total packet drop rate of 6% and SDN-Guard has a total packet drop rate of 24%. This means that more packets are dropped in the case of the SDN-Guard. We, therefore, conclude that the SD-CRN is better in terms of PDR compared to the SDN-Guard.

4.7. Round Trip Time

RTT is the sum of the times it takes for a data packet to be sent to a destination and for an acknowledgment to be received by the sender. This metric is important in measuring how fast data is transmitted when a DoS attack is launched with our scheme mitigating the attack.

Figure 13 shows the performance of RTT with and without SD-CRN Shield.

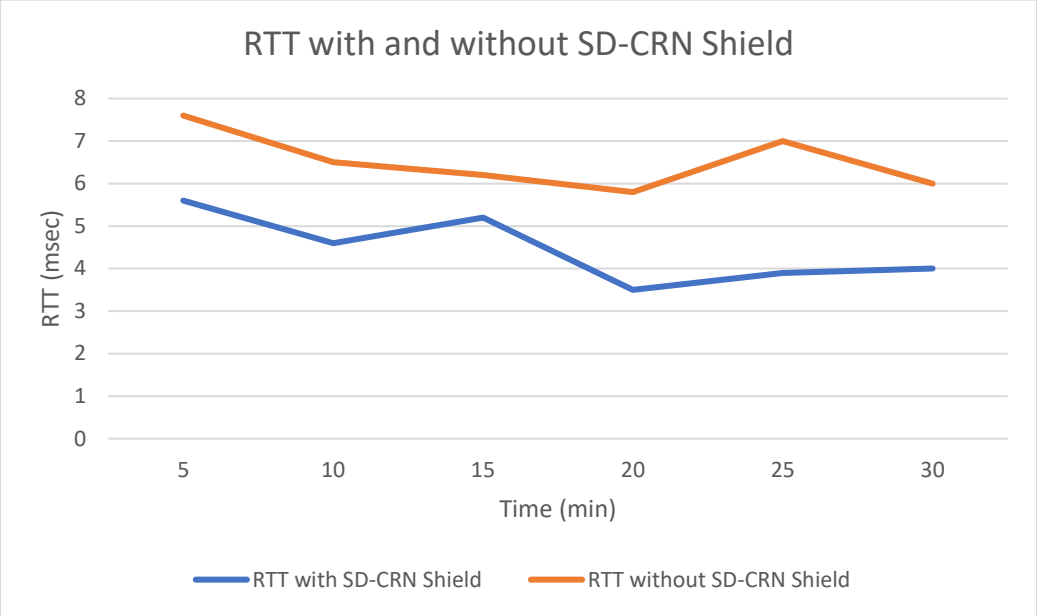

Figure 13: RTT with and without SD-CRN Shield

When the simulation is run without the SD-CRN Shield, we observe that the RRT is high. This means that nodes take a long time to send and receive data because packets are lost during the DoS attack and there is no layer of protection to prevent the packets from being dropped.

When the simulation is run with the SD-CRN Shield, we observe that the RTT has reduced. This means that our scheme is able to reduce the RTT when there is a DoS attack, confirming that this scheme is effective. We then compared the performance of SD-CRN Shield to the SDN-Guard based on the RTT metric in Figure 14.

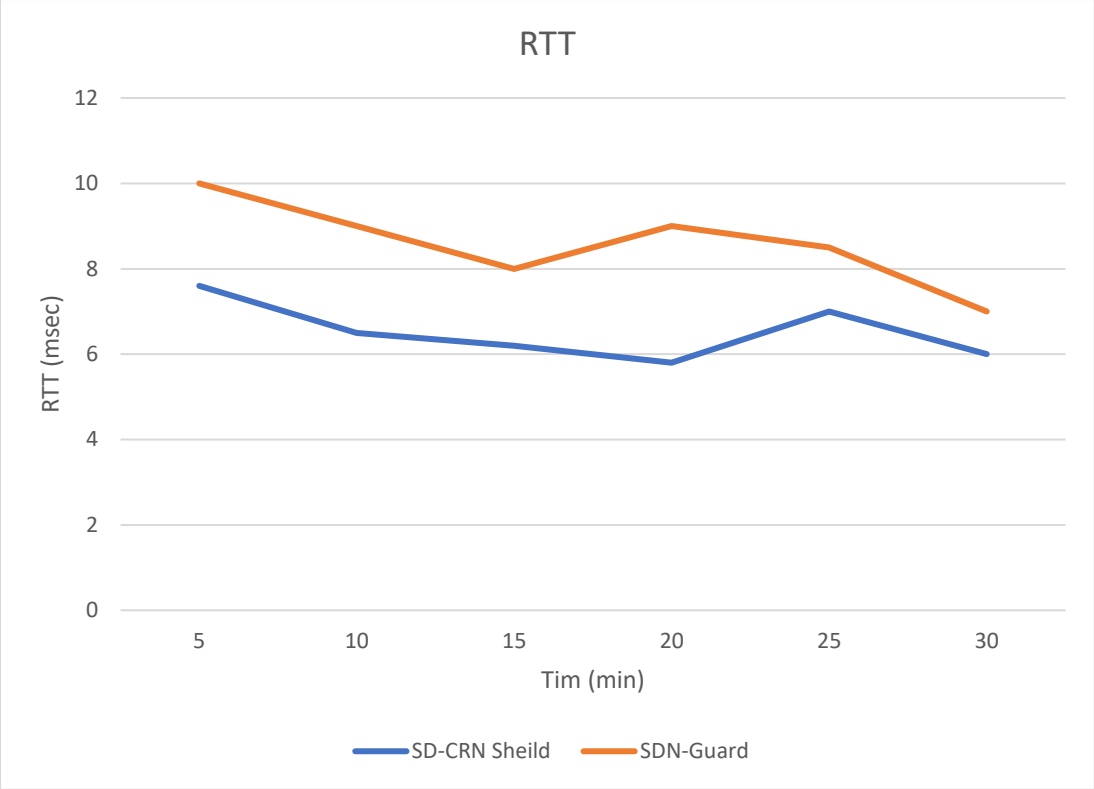

Figure 14: Comparative RTT Results

The figure demonstrates that the SDN-Guard RTT is higher than that of the SD-CRN Shield, meaning that when the attack is launched, the SD-CRN Shield takes less time to send and receive packets. We can therefore conclude our scheme is superior to the SDN-Guard.

4.8. Payload

The payload is the part of transmitted data and consists of the signal and control related data of the packets. In the context of a DoS attack, the payload is the portion of malware that the attacker intends to deliver to the victim. We compared the payload of the SD-CRN Shield to the one of the SDN-Guard in Figure 15.

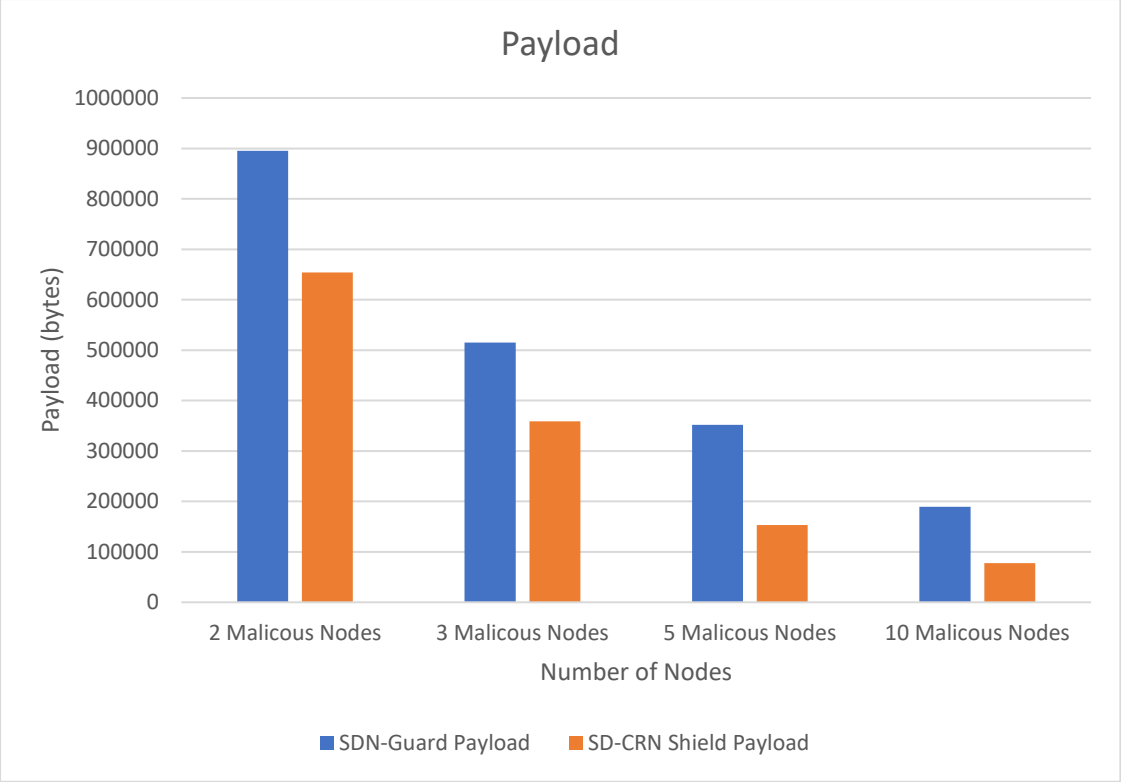

Figure 15: The Analysis of SD-CRN Shield and SDN-Guard Payload Results

The results illustrate that the payload of the SDN-Guard is greater than the payload of the SD-CRN Shield in all the cases, meaning that the network is more susceptible to DoS attacks with the SDN-Guard. The SD-CRN Shield was able to reduce the payload when we have 2, 3, 5, and 10 malicious nodes, verifying that the SD-CRN Shield is better in terms of reducing DoS attacks in SD-CRNs.

4.9. Jitter

Jitter is the variation in the time delay between when a signal is transmitted and when it's received over a network connection, measuring the variability in ping. Jitter is used to assess the performance of the network. It is evident when there is a time delay in the sending of these data packets over a network connection. In Figure 16, we observed the Jitter of the SD-CRN Shield and compared it to the SDN-Guard.

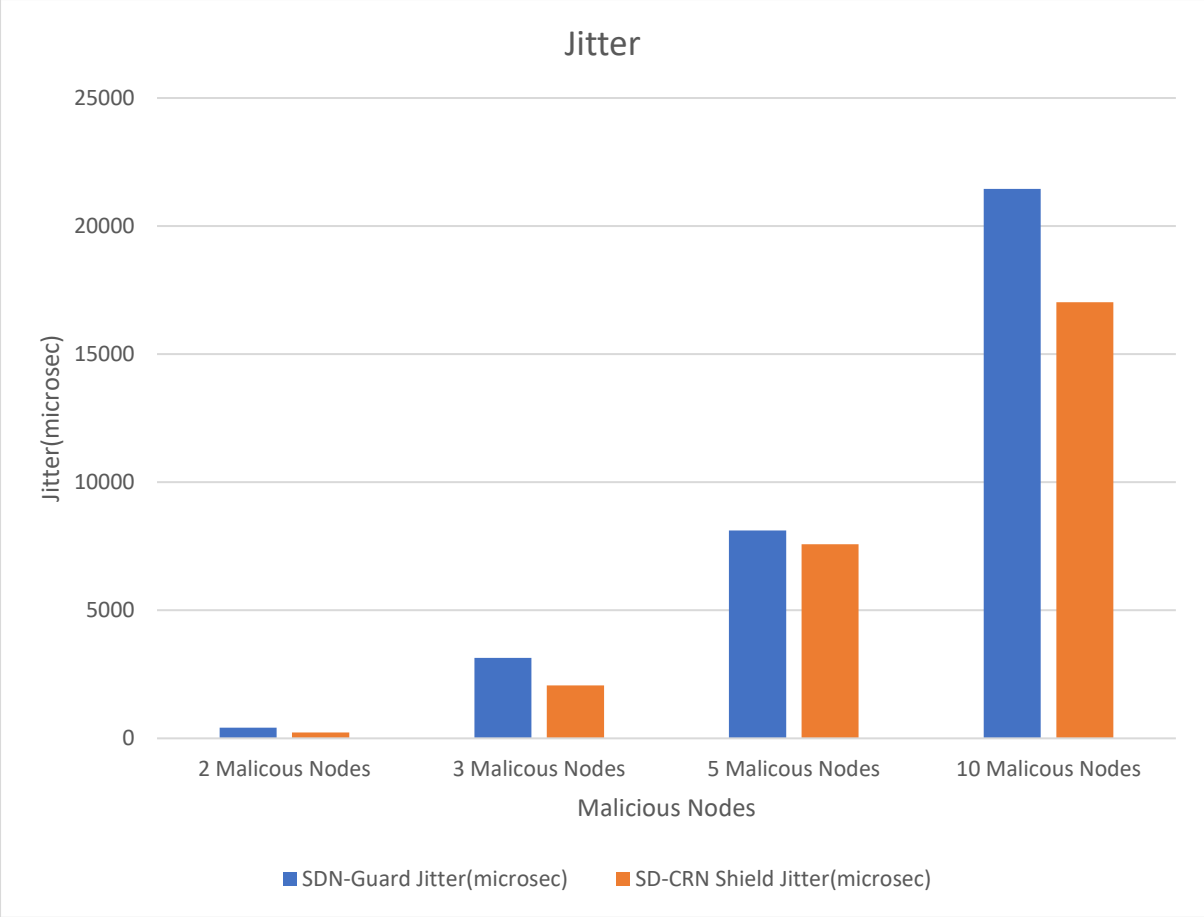

Figure 16: The Comparative Jitter Results

The results in Figure 16 show that the SD-CRN Shield has less jitter compared to the SDN-Guard in scenarios with 2, 3, 5, and 10 malicious nodes. For 2 malicious nodes, the jitter for the SDN-Guard is 411,21ms while the one for SD-CRN Shield is 233,25ms. For 3 malicious nodes, the jitter for the SDN-Guard is 3145,22ms and the jitter for SD-CRN Shield is 2072,66ms. In the scenarios with 5 malicious nodes, the jitter for the SDN-Guard is 8111,2ms and the SD-CRN Shield is 7578ms while in the one with 10 malicious nodes, the jitter for the SDN-Guard is 21446,54ms and SD-CRN Shield is 17021,15ms. This shows that the SD-CRN Shield is effective in addressing the effects of DoS attacks.

It is evident that the SD-CRN Shield achieved better jitter results, making the SD-CRN Shield a better scheme because a high jitter level is not good for network reliability. The scheme is also effective in addressing the effects of DoS.

4.10. Detection Time

Detection time is the period it takes to identify that an attack has occurred. We evaluated the detection times of the SD-CRN Shield and the SDN-Guard because early detection of an attack is desirable as a quality-of-service metric. The detection time results are shown in Figure 17.

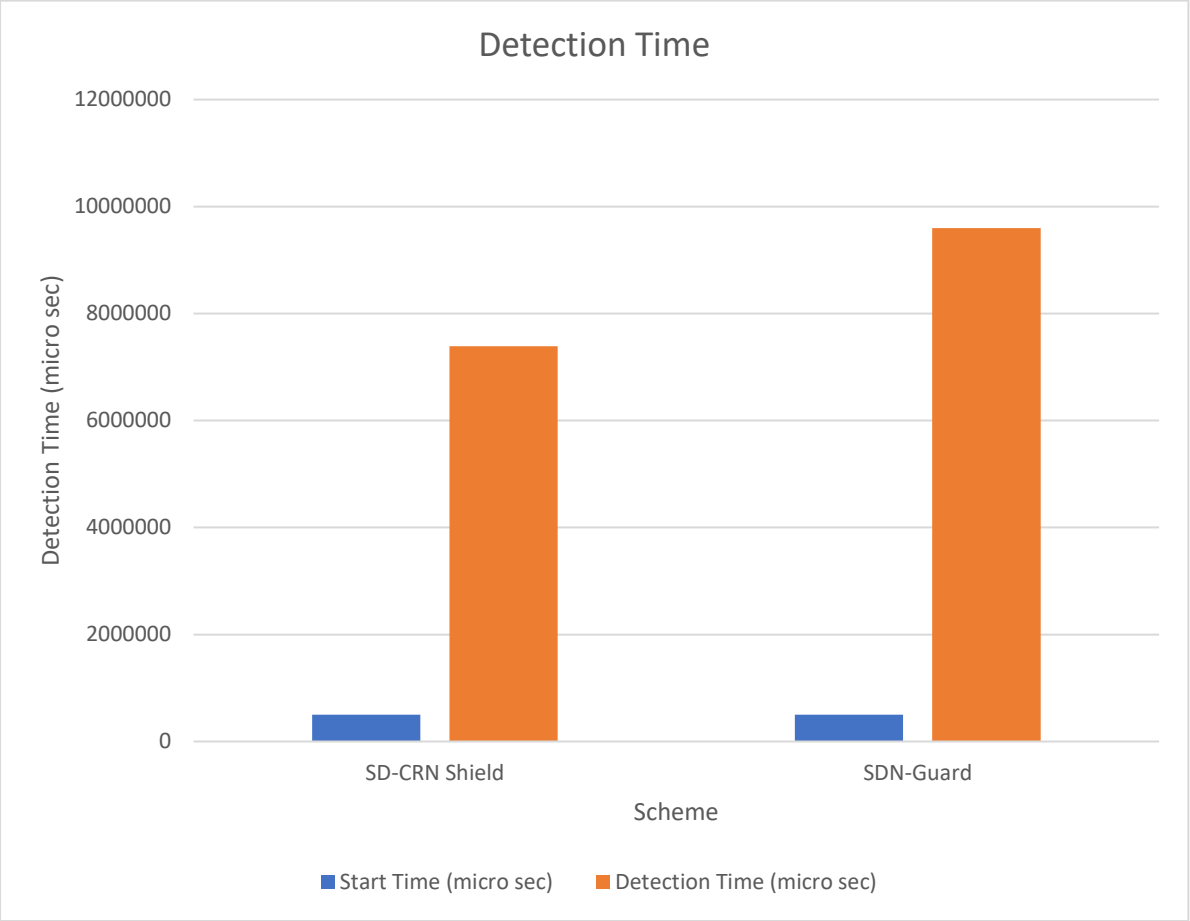

Figure 17: Investigating the Detection Times of the two Schemes

The results show the Start Time and the Detection Time of the DoS attack of the SD-CRN Shield and the SDN-Guard. The start time of the attack is 500000μs, which is equal to 5 seconds, for both schemes. The detection time for the SD-CRN Shield is 7386986,02μs, which is equal to 7,38698602 seconds. and the detection time for the SDN-Guard is 9595413,82μs, which is equal to 9,59541382 seconds. It is clear that the SD-CRN Shield detects the attack earlier compared to the SDN-Guard. We

therefore conclude that the SD-CRN Shield scheme holds a higher detection rate coupled with earlier detection time.

4.11. Summary

This chapter presented the results of the simulations and the different scenarios considered in this study. The results generated were also interpreted and discussed. We evaluated the results of our scheme, the SD-CRN Shield and compared its performance to the SDN-Guard. We evaluated the throughput, PDR, RTT, payload, jitter, and detection time results of the two schemes. The comparative results show that our scheme mitigates the effects of the DoS attack in SD-CRN under different scenarios more effectively than the SDN-Guard. This shows that the SD-CRN Shield is superior.

CHAPTER 5 – CONCLUSION

5.1 Introduction

The adoption of SD-CRN software is on the rise, therefore resolving its security issues is becoming more popular than before. We focused specifically on DoS attacks in this study. DoS attacks are meant to shut down a machine or a network, making it inaccessible for the legitimate users. In our research we simulated SD-CRN and a DoS attack in the network. We then designed the SD-CRN Shield to detect and mitigate the Dos attack. To measure the efficiency of the SD-CRN Shield, we compared it against the SDN-Guard.

As we conclude the study, we summarize our results and emphasize the research's significance. Moreover, we recommend future research that can build on the SD-CRN Shield scheme's benefits while addressing its shortcomings in order to improve the security of SD-CRN.

5.2. Research summary

Our study described recent developments in SD-CRN security. For this study, we designed the SD-CRN Shield scheme and demonstrated the various SD-CRN techniques. We aimed to design an SD-CRN Shield that detects and mitigates DoS attacks in SD-CRNs. We implemented 5 scenarios where we tested the SD-CRN Shields performance and compared it to that of the SDN-Guard. The first scenario was no malicious nodes, the second was 1 malicious node, the third was 2 malicious nodes, the fourth scenario was 5 malicious nodes, and the fifth scenario was with 10 malicious nodes. The analysis of the simulation results showed that our scheme, SD-CRN Shield, performed significantly well given the different scenarios using the throughput, PDR, jitter, payload, detection time and RTT.

Throughout this study, we have emphasized the importance of DoS attacks in SD-CRN and the importance for a comprehensive security solution which is capable of detecting and mitigating DoS attacks in real time. The SD-CRN Shield presented in this study is a promising approach that employs cognitive radio and software-defined networking techniques to produce an intelligent network shield capable of detecting and mitigating DoS attacks while protecting legitimate users.

5.3. Recommendations

DoS is a major security threat in SD-CRN. It affects the performance of the entire CRN. Hence, there is a need for the simplest technique to detect abnormal changes in the network. Secondary users should access the spectrum without causing interference to the primary user. Furthermore, IDS for other types of attacks could also be studied in forthcoming studies. In the future, the SD-CRN Shield could be improved in terms of detection latency and detection capability. In addition, it could be compared to the other schemes. Other metrics such as CPU Usage and Threat Detection Time could be used to effectively compare the schemes in the future.

While our experimental findings suggest that the SD-CRN Shield scheme is successful, there is still space for improvement. Future research may concentrate on

improving the performance of the scheme's machine learning algorithms, improving the accuracy of DoS detection, and reducing false positives. In addition, to provide a comprehensive security solution for SD-CRN, the scheme can be enhanced to incorporate other security measures such as intrusion prevention systems (IPS) and firewalls.

5.4. Conclusion

In our study, we created a scheme to reduce and detect DoS attacks in SD-CRN. The SD-CRN Shield relies on an IDS that constantly communicates with it to inform it of any irregularities. Our simulation metrics were throughput, round trip time, and packet drop rate, jitter, payload, and detection time. Our scheme achieved less throughput and packet drop rate, and lesser round-trip time. Our scheme showed a faster detection time and lower jitter. The payload of our scheme was less compared to the SDN-Guard.

Our results have demonstrated that the SD-CRN Shield is an effective scheme for detecting and mitigating DoS attacks in SD-CRN. When a DoS attack is identified, SD-CRN Shield can change the network parameters to counter the effects of the attack by dynamically re-routing traffic, filtering malicious data, or slowing down traffic flows.

The SD-CRN Shield reduces DoS attacks in SD-CRN. With more improvements, the scheme has the potential to providing an effective and efficient protection.

References

[1] M. K. Y. Jararweh, "SD-CRN: Software Defined Cognitive Radio Network Framework," in *2014 IEEE International Conference on Cloud Engineering*, 2014.

[2] L. Dridi and M. F. Zhani, "SDN-Guard: DoS Attacks Mitigation in SDN," in *Ecole de Technologie Superieure(ETS)*, Canada, 2016.

[3] V. Tai and S. Sengupta, "DoS: Attack and Defense".

[4] V. Durcekova, L. Schwartz and N. Shahmehri, "Sophisticated Denial of Service Attacks aimed at Application Layer," in *Elektro 2012*, 2012.

[5] S. Kumar, M. Sachdeva and . D. K. Kumar, "Flooding Based DDoS Attacks and Their Influence on Web Services," *International Journal of Computer Science and Information Technologies,* vol. 2, pp. 1131-1136, 2011.

[6] C. Douligeris and A. Mitrokotsa, "DDoS attacks and defense mechanisms: classification and state-of-the-art," *Computer Networks,* vol. 44, no. 5, pp. 643-666, 2004.

[7] A. Kuzmanovic and E. W. Knightly, "Low-Rate TCP-Targeted Denial of Service Attacks and Counter Strategies," *IEEE/ACM TRANSACTIONS ON NETWORKING,* vol. 14, no. 4, 2006.

[8] M. Imran, M. D. Hanif, F. A. Khan and A. Derhab, "Reducing the effects of DoS attacks in software-defined works using parallel for installation," Saudi Arabia, 2019.

[9] R. Kloti, V. Kotronis, and P. Smith, "OpenFlow: A Security Analysis," in *21st IEEE international conference on the network*, New York, 2013.

[10] E. J. Leavline, M. Dinesh and D. A. A. G. Singh, "Jamming Attack Detection Technique in Cognitive Radio," *International Journal of Applied Engineering Research,* vol. 10, no. 55, pp. 2347 - 2812, 2015.

[11] P. Zhang, H. Wang, C. Hu, and C. Lin, "On Denial of Service Attacks in Software Defined Networks," *IEEE Network,* vol. 30, no. 6, pp. 28-33, 2016.

[12] L. Wei and C. Fung, "FlowRanger: A request prioritizing algorithm for controller DoS attacks in software-defined networks," in *2015 IEEE International Conference on Communications (ICC)*, London, UK, 2015.

[13] S. S. Hayward, S. Natarajan, and S. Sezer, "A survey of security in software-defined networks," *IEEE Communications Surveys & Tutorials,* vol. 18, pp. 623-654, 2016.

[14] I. Ahmad, S. Namal and M. Ylianttila, "Security in sofware defned networks: a survey," *IEEE Communications Surveys& IEEE Communications Surveys&,* vol. 17, pp. 2317-2346, 2015.

[15] Q. Yan and F. R. Yu, "Distributed denial of service attacks in software-defined networking with cloud computing," *IEEE Communications Magazine,* vol. 53, pp. 52-59, 2015.

[16] H. Wang, L. Xu, and G. Gu, "FloodGuard: A DoS Attack Prevention Extension in Software-Defined Networks," in *Proc IEEE/IFIP Int'l. Conf. Dependable Systems and Networks*, 2015.

[17] S. Shin, "AVANT-GUARD: Scalable and Vigilant Switch Flow Management in Software-Defined Networks," in *Proc. ACM CCS*, 2013.

[18] K. Wolter and Reineck, "Performance and security tradeoff.," In International School on Formal Methods for the Design of Computer, Communication and Software System, Springer Berlin Heidelberg, 2010.

[19] T. Ubale and J. A. Kumar, "SRL: An TCP SYNFLOOD DDoS Mitigation Approach in Software-Defined Networks," in *Proceedings of the 2nd International Conference on Electronics, Communication and Aerospace Technology*, 2018.

[20] M. Z. A. Aziz and K. Okamura, "Leveraging SDN for Detection and Mitigation SMTP Flood Attack through Deep Learning Analysis Techniques," *International Journal of Computer Science and Network Security,* vol. 17, no. 10, pp. 166-172, 2017.

[21] M. Imran, M. H. Durad, F. A. Khan and A. Derhab, "Reducing the effects of DoS attacks in software-defined networks using parallel flow," in *Hum. Cent. Comput. Inf. Sci*, 2019.

[22] T. Wang, H. Chen, G. Cheng, and Y. Lu, "SDNManager: A Safeguard Architecture for SDN DoS Attacks Based on Bandwidth Prediction," *Security and Communication Networks,* p. 16, 24 November 2017.

[23] C. Ejike and D. Kouvatsos, "Detection of Network Congestion and Denial of Service (DoS) Attacks in Cognitive Radio Networks," in *2019 7th International Conference on Future Internet of Things and Cloud*, 2019.

[24] P. V. R. Saravanan, "A Comprehensive Study on DoS Attacks and Countermeasures in Software-Defined CRNs," *Wireless Personal Communications.,* vol. 121, pp. 2273-2290, 2021.

[25] K. Z. W. Liu, "Analysis of Denial of Service Attacks and Their Mitigation Techniques in Software Defined Cognitive Radio Networks," *IEEE Access,* 2021.

[26] S. Kotb, H. El-Dien, and A. Eldien, "Machine learning-based Distributed Denial-of-Service Detection Scheme for," in *International Mobile, Intelligent, and Ubiquitous Computing Conference*, Cairo, Egypt, 2021.

[27] Y. Jararweh, A.-A. Mahmoud, A. Doulat, A. A. Aziz, H. B. Salameh, and A. Khreishah, "SD-CRN: Software Defined Cognitive Radio Network Framework," in *2014 IEEE International Conference on Cloud Engineering*, 2014.

[28] H. Wang, D. Zhang, and K. G. Shin, "Change-Point Monitoring for Detection of DoS Attacks," *IEEE Trans. on Dependable and Secure Computing,* vol. 1, no. 4, pp. 193-208, 2004.

[29] M. Zubair, M. Fouda, . H. Nishiyama and N. Kato, "Intrusion Detection System (IDS) for Combating Attacks Against CognitiveRadio Networks," *IEEE Network Magazine,* vol. 27, no. 3, pp. 51-56, 2013.